

## PROBLEM. Matrix of Strings (contributed by J. Beredo) [Score: 50/50]

As per the Mathematics definition, a matrix is an arrangement of numbers, symbols, or letters in rows and columns which is used in solving mathematical problems <sup>1</sup>. You can perform different types of operations with these groups of values. For this exam, we will be storing strings and performing operations on a **square matrix** - a matrix with the same row and column size.

To complete this exam, you are to implement five (5) functions that will allow your program to:

1. Accept string inputs given an input size then store to a 2D array (5 pts)
2. Transpose the string matrix and save to another array (5 pts)
3. Store the reverse of all the string inputs (10 pts)
4. Copy all strings that are palindromes into a 1D array (15 pts)
5. Copy all strings with more vowels than consonants (15 pts)

**TAKE NOTE:** All display functions are called inside the main function.

Refer below for examples of expected results for the features indicated above. The examples are using a size of 3 (that is, 3 rows and 3 columns), but note that the input of the user for size can be at most 10 (equal to the defined constant SIZE).

### TRANSPOSING

The transpose of the matrix is created by swapping the rows to columns or columns to rows.

*Example:*

```
ORIGINAL STRINGS
lynx      eve      andromeda
auriga    cygnus   civic
aquarius  aviva     hydra

TRANPOSED MATRIX
lynx      auriga    aquarius
eve       cygnus    aviva
andromeda civic     hydra
```

### REVERSING

For the reverse function in this problem, each string in the original matrix should be reversed.

*Example:*

```
ORIGINAL STRINGS
lynx      eve      andromeda
auriga    cygnus   civic
aquarius  aviva     hydra

REVERSED STRINGS
xnyl      eve      ademordna
agirua    sungyc   civic
suirauqa  aviva     ardyh
```

---

<sup>1</sup> <https://www.collinsdictionary.com/dictionary/english/matrix>

## PALINDROMES

A *palindrome* is a word, sentence, verse, or even number that reads the same backward or forward. Assume the stored string only contains small letters (no special characters).

*Example:*

```
ORIGINAL STRINGS
lynx      eve      andromeda
auriga    cygnus   civic
aquarius  aviva     hydra

PALINDROMES
1 eve
2 civic
3 aviva

TOTAL COUNT: 3
```

## VOWELED WORDS

Voweled words are those words having more vowels than consonants. Assume the stored string only contains small letters. In case of equal vowels and consonants, exclude the word.

*Example:*

```
Row and column size:
ORIGINAL STRINGS
lynx      eve      andromeda
aurigans  cygnus   civic
aquarius  aviva     hydra

VOWELED STRINGS
1 eve
2 aquarius
3 aviva
```

## EXAMPLE COMPLETE PROGRAM RUNS (assuming size input is 3 as in input.txt)

```
Row and column size:
ORIGINAL STRINGS
lynx      eve      andromeda
aurigans  cygnus   civic
aquarius  aviva     hydra

TRANPOSED MATRIX
lynx      aurigans  aquarius
eve       cygnus   aviva
andromeda civic     hydra

REVERSED STRINGS
xnyl      eve      ademordna
snagirua  sungyc  civic
suirauqa  aviva   ardyh

PALINDROMES
1 eve
2 civic
3 aviva

TOTAL COUNT: 3

VOWELED STRINGS
1 eve
2 aquarius
3 aviva
```

```
Row and column size:
ORIGINAL STRINGS
mom      father
uncle    auntie

TRANPOSED MATRIX
mom      uncle
father   auntie

REVERSED STRINGS
mom      rehtaf
elcnu    eitnua

PALINDROMES
1 mom

TOTAL COUNT: 1

VOWELED STRINGS
1 auntie
```

To answer this programming problem, you are given the following seven (7) accompanying files:

- matrix.h - the header file that contains the macro definitions and function prototypes that you need to define (complete) in LASTNAME-Matrix.c file (DO NOT CHANGE ANYTHING IN THIS FILE)
- LASTNAME-MATRIX.c - the skeleton file where you will encode your functions
- main.c - the C file containing main() function with function calls to the five functions you have to complete and display functions to show the results (DO NOT CHANGE ANYTHING IN THIS FILE)
- input.txt - a text file containing a sample COMPLETE input
- Output.txt - a text file containing the expected output of the program given the input.txt
- input2.txt - second sample text file containing a sample COMPLETE input
- output2.txt - second text file containing the expected output of the program given the input2.txt

### **RUN YOUR PROGRAM WITH INPUT and OUTPUT REDIRECTION:**

Run your exe file in the command line with input redirection. You can store the result of your program into a text file by output redirection, For example, if **after compiling main.c**, you have the executable **prog.exe**, then you can run your executable program with output redirection as:

```
C:\CCPROG2> prog < input.txt > SANTOS-ACTUAL.txt
```

The output of the exe file will be stored in SANTOS-ACTUAL.txt file. A correct program using the input.txt for input redirection should produce the same set of values that are in output.txt provided.

### **TESTING & SCORING:**

- Your program will be compiled via gcc -Wall, using C99 standard. Thus, for each function that does not compile successfully, the score for that function is 0.
- Your program will be tested by your instructor with other TEXT FILES (contents are different from the ones given to you) and with function calls of different parameter values.
- Full credit will be given for the function only if the student's implementation is all correct for all the test values used by the instructor during checking AND only if the student's implementation complied with the requirement and did not violate restrictions. Brute force solutions will not be given any credit. Deductions will be given if not all test cases produce correct results OR if restrictions were not followed.

### **DELIVERABLES:**

Submit/upload two files via Canvas before the indicated deadline:

1. your LASTNAME-MATRIX.c source file
2. your LASTNAME-ACTUAL.txt output text file

*Don't forget to change the filenames to your own last name.*

**-- THE END --**